https://doi.org/10.20965/jrm.2021.p1303

Paper:

Development of an Automatic Tracking Camera System Integrating Image Processing and Machine Learning

Masato Fujitake*, Makito Inoue*, and Takashi Yoshimi**

*Graduate School of Engineering and Science, Shibaura Institute of Technology
3-7-5 Toyosu, Koto-ku, Tokyo 135-8548, Japan
E-mail: {MA17097, MA17017}@shibaura-it.ac.jp
**College of Engineering, Shibaura Institute of Technology
3-7-5 Toyosu, Koto-ku, Tokyo 135-8548, Japan
E-mail: yoshimit@shibaura-it.ac.jp
[Received June 11, 2021; accepted September 7, 2021]

This paper describes the development of a robust object tracking system that combines detection methods based on image processing and machine learning for automatic construction machine tracking cameras at unmanned construction sites. In recent years, unmanned construction technology has been developed to prevent secondary disasters from harming workers in hazardous areas. There are surveillance cameras on disaster sites that monitor the environment and movements of construction machines. By watching footage from the surveillance cameras, machine operators can control the construction machines from a safe remote site. However, to control surveillance cameras to follow the target machines, camera operators are also required to work next to machine operators. To improve efficiency, an automatic tracking camera system for construction machines is required. We propose a robust and scalable object tracking system and robust object detection algorithm, and present an accurate and robust tracking system for construction machines by integrating these two methods. Our proposed image-processing algorithm is able to continue tracking for a longer period than previous methods, and the proposed object detection method using machine learning detects machines robustly by focusing on their component parts of the target objects. Evaluations in real-world field scenarios demonstrate that our methods are more accurate and robust than existing off-the-shelf object tracking algorithms while maintaining practical real-time processing performance.

Keywords: object detection, image processing, unmanned construction, machine learning

1. Introduction

Many natural disasters such as earthquakes, volcanic eruptions, and sediment-related disasters occur frequently in Japan. For example, severe damage occurred in the Kumamoto prefecture with numerous structures collapsing during a 2016 earthquake. It is important to prepare for these natural disasters. However, it is also important to deal with hazards after a primary catastrophic disaster occurs to minimize additional damage. Reconstruction work such as removing debris faces the potential hazards of secondary disasters, and it is dangerous for workers to operate in hazardous areas. For these reasons, the research and development of unmanned construction technology that allows workers to monitor and operate construction machines from a safe remote operation site has been conducted and this technology has been applied at real worksites to avoid danger in recent years [1–7]. There are two main approaches to realizing unmanned construction. The first is to use autonomous construction machines that are completely independent of worker command and the second is to use semi-autonomous construction machines that workers operate remotely. Currently, the main method of unmanned construction is to operate semi-autonomous construction equipment remotely because autonomous construction equipment is still under development. In unmanned construction with semi-autonomous construction machines, the workers who operate construction machines from remote sites depend on real video images captured by surveillance cameras installed at the worksite. The viewpoint of the surveillance cameras in this method is a third-person viewpoint, so both the construction machines and surrounding environment are reflected in the images. Therefore, it is easy for operators to grasp the positional relationships between construction equipment and surrounding objects. A surveillance camera and its platform must be controlled to capture construction machines continuously. The surveillance cameras in disaster areas are controlled by camera operators, rather than construction machine operators. A surveillance camera operator manages multiple devices simultaneously. To improve the efficiency of surveillance camera operator work, the development of an automatic tracking camera system for construction machines is required for unmanned construction sites.

Generally, to track a target construction machine automatically at an unmanned construction site, there are two

main steps. The first step is to detect construction machines in images captured by surveillance cameras. In the second step, the tracking system turns a camera toward the target by controlling the corresponding camera platform. Detecting construction machines robustly is an essential challenge for constructing an autonomous tracking camera system. However, this is very difficult because the appearance of construction machines changes drastically as they work and/or the weather changes.

Algorithms for detecting specific objects in an image have been studied for a long time in the field of object detection in computer vision [8]. Object detection methods can be roughly divided into two types. In the first type, there are several simple methods such as template matching and color extraction [9]. These approaches to detecting objects are computationally efficient, but struggle to recognize various changes in object appearance. Regarding the second type, some complex algorithms based on machine learning have been studied [10, 11]. In comparison to simple methods, these complex methods have more robust detection abilities, but they require a large number of computations. Even if their detection ability is high, if the detection procedure takes too much time, there could be a difference between the estimated and actual positions of an object such as construction machine that is moving.

Therefore, the technical challenges associated with the target problem are twofold. On the algorithm side, we must design object tracking methods that are robust to changes in object appearance. On the system side, we must design practical combinations of available hardware and software components for multiple surveillance cameras. Therefore, by improving rapid image processing performance using simple methods and achieving robust object detection using machine learning, and combining these approaches appropriately, we propose a novel robust and accurate object tracking algorithm and system. The developed automatic tracking camera system (ATCS) can capture a target machine robustly, even if it moves quickly and turns. In this study, we tested the proposed system with one surveillance camera, but it is easy to add another camera to the system. Section 2 describes the proposed algorithm and structure of the proposed system. Experiments using the proposed object detection methods called ATM3D (automated template matching meets motion detection) based on image processing and POLO (part-based YOLO) based on machine learning are described in Section 3. Experiments to demonstrate the robustness of the proposed system in the field are presented in Section 4 and an improvement of the proposed integration of the two methods is discussed in Section 5. The goal for the proposed method and system is application to unmanned construction sites. However, they are a highly versatile method and system in which the camera follows a moving object automatically and they are not limited to this field. We expect our method to be applied to other fields such as automatic monitoring systems in the future.

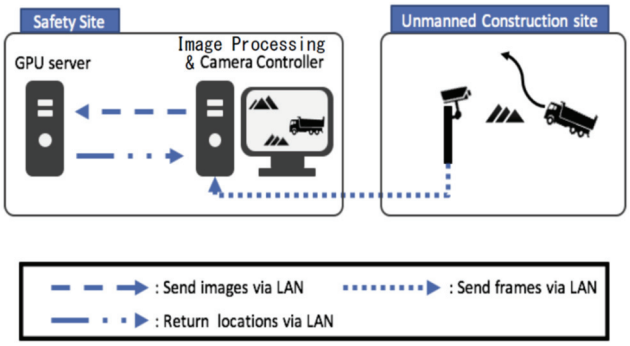


Fig. 1. Design of the ATCS architecture.

2. Design of the ATCS Architecture

2.1. Overview of the System

The proposed system consists of three components: a camera module, computer for simple image processing and camera platform control, and computer for machine learning. The design of the ATCS is illustrated in **Fig. 1**. The camera module is installed at an unmanned construction site to monitor the activities of construction machines and other changing information. It provides a series of images of the construction site to the computers for image processing. In this system, we used an internet protocol camera model AXIS Q1765-LE, which is a digital video camera, and sent data through a computer network. The camera is mounted on a camera platform that receives control commands from the simple image processing and camera platform control computer through the network to track the target construction machine. Therefore, we can safely change the direction of the camera installed at a construction site. The simple image processing and camera platform control computer receives data from the camera and sends commands to control the direction of the camera. The computers apply the proposed image-processing algorithm and recognize target construction machines.

The ATCS is required to control the direction of the camera to follow moving construction machines and avoid deviating from the monitoring screen at each worksite. Generally, one camera is used to monitor a construction machine that moves around an area of up to 80×80 to 100×100 m at a maximum speed of approximately 10–15 km/h at an unmanned construction site. If the working area is wider, then we install multiple cameras at an unmanned construction site.

In our system, we combine two types of image processing: a simple method that attempts to track target objects, but does not understand what the objects are, and the complex method that learns the appearance of target objects and detects them. The simple algorithm combines two image processing methods: a template matching method and motion detection. In contrast, the complex method exploits YOLO networks, which are well-known neural networks for object detection based on convolutional neural networks (CNNs) [12, 13]. This method detects ob-

ject parts and integrates their information to identify a complete object. We propose a simple algorithm called ATM3D and complex algorithm called POLO. ATM3D is a lightweight image processing method used to track an object and is executed on the same computer as the camera platform control, although it may fail to continue tracking objects. For example, it sometimes fails to track when the appearance of a construction machine changes drastically as a result of turning. In contrast, POLO is a heavyweight image processing method based on machine learning that detects target objects more robustly. Therefore, by combining ATM3D and POLO and using both results, it is possible to meet the requirements of the system in terms of both the recognition ability and processing speed of the target object. The details of ATM3D and POLO are discussed in Sections 2.2 and 2.3, respectively. It is difficult to execute POLO on the same computer executing ATM3D because it requires a long time for its calculations. Therefore, we use a GPU-accelerated computer for POLO to improve processing speed.

The simple image processing and camera platform control computer sends images provided by the surveillance camera to the machine learning algorithm executing computer through the internet. In our initial experiments, the accelerated computer that executes POLO returned recognition results at approximately 4 fps over a mobile network. Because the GPU server executes this time-consuming process, the simple image processing and camera platform control computer only needs to process lightweight calculations. A system design using these two separated algorithms reduces the cost of the overall system and enables the easy addition of another camera. The details of the integration process for these two algorithms are described in Section 2.4.

2.2. Detection Algorithm Based on Image Processing

In this section, we introduce the design of the lightweight image processing algorithm ATM3D for tracking an object [14]. The cornerstone of this algorithm is the combination of two types of image processing methods: template matching and background subtraction.

Template matching is an algorithm for finding a small area in an image that matches a template image from the field of computer vision. By using this method to detect construction machines, the area corresponding to the region of a target machine can be identified if a template image that reflects the appearance or parts of the construction machine is similar to a specific part of an image. However, it is difficult to detect the region of an object continuously when the image of the target object changes significantly. Therefore, we improved the tracking method to update the template image automatically in real time. Specifically, a new template image is generated from the area that was detected using the previous template image (**Fig. 2**). This operation is performed on each frame. This auto-update template matching method enables the system to track a target object more robustly

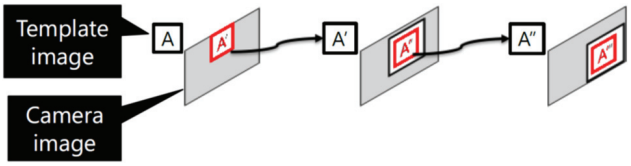


Fig. 2. Auto update template matching method.

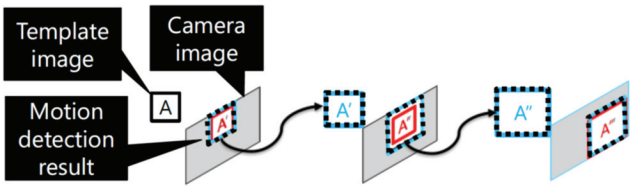


Fig. 3. Motion detection method.

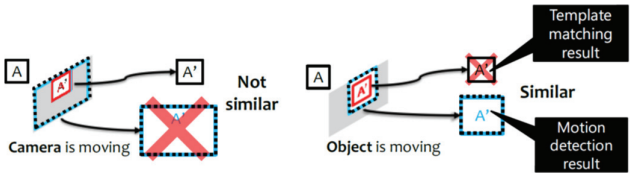


Fig. 4. Developed ATM3D algorithm.

than the original method, even when a construction machine changes its orientation.

Background subtraction is an algorithm for extracting different areas of two images from the field of image processing. It is known as a motion detection method. This method is often used for video streaming applications with static cameras to find regions in which something moves [15]. We generate a new template image from the moving area detected by the background subtraction method (**Fig. 3**). The detection of construction machines via background subtraction is robust to problems such as appearance changes of target machines caused by turning, whereas the template matching method is susceptible to this type of issue. However, when the camera moves and the entire screen changes, it is difficult to detect moving objects.

We developed an algorithm called ATM3D that combines the two methods outlined above to track target objects robustly and overcome the issues of the auto-update template matching method and background subtraction method. ATM3D compares the results of the two methods and if both results are similar, then the system adopts the detected area from the background subtraction method as the next template image (**Fig. 4**). In other words, when the output regions of both methods are similar, the results of moving object detection are used preferentially as the next template image. The degree of similarity between two results is calculated using the Jaccard similarity coefficient, which is also known as intersection over union (IoU) [16].

Fig. 5. Difficult scenario for detecting a construction machine.

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|}. \quad \ldots\ldots\ldots\ldots \quad (1)$$

Here, A and B represent the detected regions.

ATM3D has three significant advantages: it can follow machine appearance changes, it has good adjustability that provides variable region sizes for template images, and it has excellent stability because each method compensates for the disadvantages of the other.

2.3. Detection Algorithm Based on Machine Learning

In this section, we provide a detailed description of the proposed algorithm based on machine learning, which detects objects (i.e., construction equipment) directly. We propose an object detection method that identifies parts of an object, unlike previous object detection methods using machine learning [17].

In recent years, many machine learning methods in computer vision, particularly deep CNNs [10, 11], have been developed and have provided a series of breakthroughs for object detection and image classification [12, 18–20]. A CNN learns and obtains better target image features by updating its parameters through training iterations. Mainstream methods for object detection and tracking aim to obtain more accurate object locations by identifying target objects of various sizes in an image with high accuracy. Machine-learning-based algorithms provide some solutions to overcome the obstacles faced by traditional simple image-processing-based object detection methods. For example, machine-learning-based algorithms can detect arbitrary sizes of objects. In contrast, image processing without machine learning such as the template matching method is only suitable for fixed sizes of objects. Off-the-shelf algorithms based on neural networks work well as long as sufficient training data are available.

Another problem occurs when machine-learning-based algorithms are applied to surveillance camera tasks at unmanned construction sites. In surveillance camera detection, unlike typical object detection tasks, an entire object that should be detected sometimes cannot be viewed as a result of occlusion caused by embankments and trees. **Fig. 5** presents a difficult case of detecting a construction machine at an unmanned construction site. It is difficult to identify the construction machine in this figure because some parts are hidden by obstacles. These occlusions prevent object detection methods from finding target objects accurately because their appearances differ from what the models learned during the training process. However, some parts of an object can be viewed, even though obstacles such as embankments and trees hide other parts of the machine, except in cases where the entire object is occluded. The key motivation for our method is that detection failures caused by occlusion can often be effectively recovered by learning how objects consist of different parts [17]. In contrast to existing tracking methods based on deep convolutional networks that only consider the appearances of complete objects, the proposed part-based object detection method combines detected part information to estimate the categories and locations of entire objects.

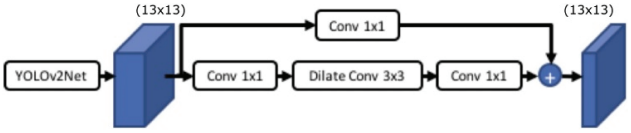


Fig. 6. Design of POLO.

Previous object-detection approaches employing CNNs are computationally expensive. It is very important for an object tracking system to detect objects quickly while maintaining accuracy. Therefore, we applied our idea to the YOLOv2 model, which is a prominent object detector [12, 13], as a baseline. This baseline model maintains competitive relative to other detectors using CNNs and it has a superior feature of detection speed. It can process images in real time at 67 fps. For one frame in a video sequence, multiple detection results may be outputted. The naive output of the YOLO model is a type of feature map that is converted into the values of results such as bounding boxes and probabilities.

We expanded the baseline model into a model that combines the results of part detection and estimates entire object locations. We call the proposed model POLO. **Fig. 6** presents the design of POLO. We use YOLOv2 as a backbone network for detecting the parts of objects and append a convolutional module that converts the feature maps of part results into a result for complete object detection. For this conversion, we exploit a bottleneck module with a shape similar to that used in ResNet [21]. The main difference between our module and ResNet is that our module replaces the convolutional operation with a dilated operation to maintain the same size of feature maps, and it uses a simple 1×1 convolution as a projection layer for extracting abstract features.

We prepared a dataset for construction machines in the format of the VOC dataset [22] containing two main classes: excavators and dump trucks. It consists of approximately 2,000 images containing dump trucks and 1,500 images containing excavators. In the dump truck dataset, there are parts such as tires, the dumping bed, and cab, which constitute the entire shape of a dump truck. In the excavator dataset, there are parts such as the

(a)

(b)

Fig. 7. Examples of annotated data.

arm, boom, bucket, counterweight, shoe, and cab, which also constitute the entire shape of an excavator. **Fig. 7** presents an example from this dataset. The black rectangles represent parts of the construction machine, such as dumping beds and arms. The white rectangles represent the entire body of the machine. In the training phase, there are two steps for training POLO. First, we use all classes, including parts such as the shoe, to train the original YOLOv2, which is partially pre-trained using ImageNet [23]. Our training procedures for YOLOv2 and POLO follow those outlined in [12, 13]. After training the model, it is possible to predict the locations of parts and their classes, including entire vehicle bodies, as shown in **Fig. 7**. Second, we append the converter module to the trained model and freeze the parameters of the trained model for all classes. We then train POLO using only the entire bodies of construction machines. As a result, POLO can detect the object surrounded by the white rectangle in **Fig. 7**. In other words, the second phase trains the weights of the converter module to transform feature maps of part information into feature maps of whole body information. To evaluate the learning process during training, we randomly divided the dataset into two groups for training and testing at a ratio of 4 : 1. Therefore, the dump trucks accounted for 64% and 58% of the training dataset and the testing dataset, respectively. The proposed model was trained using synchronized stochastic gradient descent (SGD) with a weight decay of 0.0005 and momentum of 0.9. We trained POLO using a two-step process. First, YOLOv2, which is the backbone of POLO,

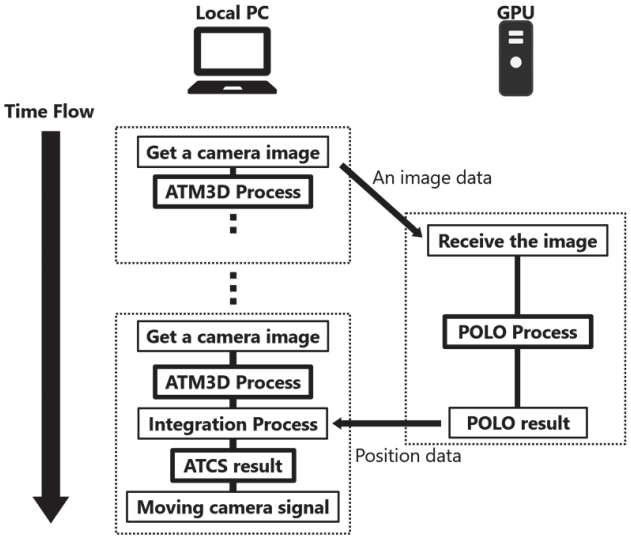


Fig. 8. Procedure of ATCS.

was loaded with pre-trained weights from ImageNet and trained to detect parts and entire objects. Therefore, the backbone neural network of YOLOv2 was initialized with the parameters of pre-trained "darknet19_448" weights, which are available from the YOLO project. For the construction machine dataset, we set the number of filters in the last convolutional layer to 80 because the total number of classes in the part detection stage was 11. Our training procedure for part detection was performed based on that described in [13]. Therefore, we also used a data augmentation method similar to that used in [13, 18]. After the part detector was trained, the converter module was appended to its final layer. The number of filters in the last layer of the converter was set to 35 to detect the complete appearances of dump trucks and excavators. We fixed the parameters of the part detector and trained the network for 40 epochs with a starting learning rate of 10^{-3}, which was divided by 10 at 30 epochs.

2.4. Integration Procedure

This section describes the steps for integrating the results of the proposed algorithms. The main procedure of the proposed system is illustrated in **Fig. 8**. The camera captures images in five steps to control the platform.

First, every image from a surveillance camera at an unmanned construction site is sent to the PC for simple image processing and camera platform control. Second, each image is processed by ATM3D for tracking construction machines and is also sent to the GPU-accelerated computer concurrently. Third, on the GPU server, POLO attempts to detect a construction machine in the image and returns a result if it is detected. While this process is executed on the GPU server, ATM3D continues to track the machine. This third step sometimes requires a small amount of additional time for the transmission of images. However, it can run at approximately 4 fps on cellular networks. Next, the simple image processing and camera platform control computer receives the outputs from the

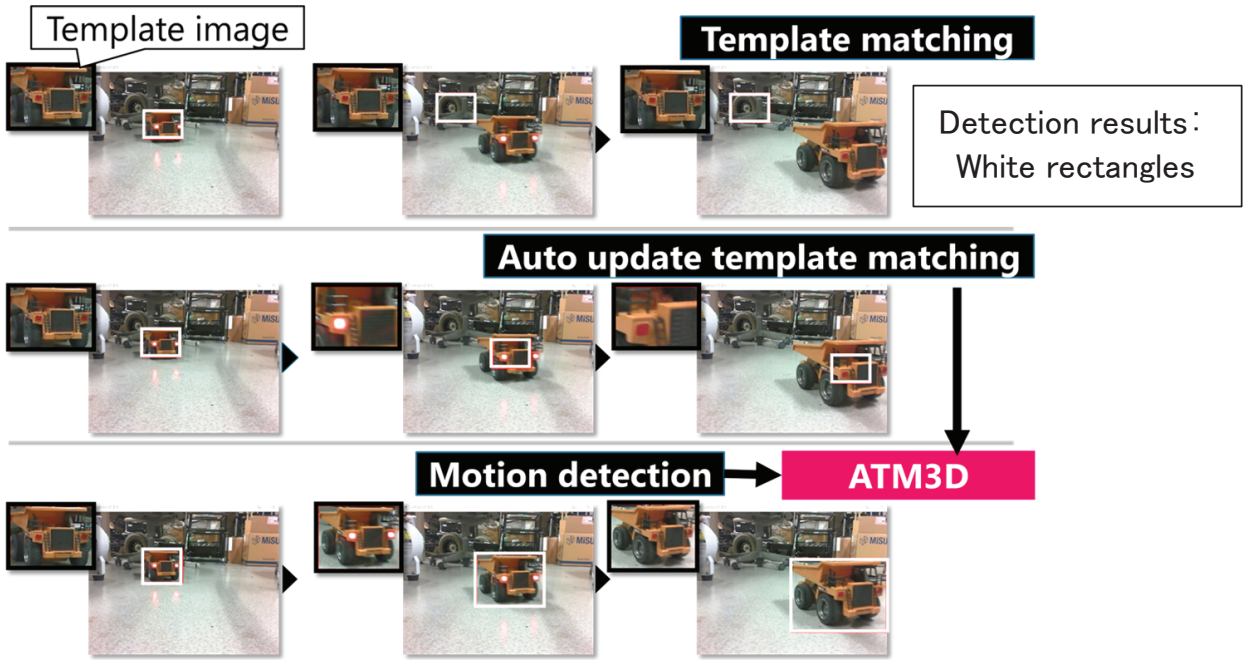


Fig. 9. Experimental results for ATM3D.

GPU server if the construction machine is identified, and compares the regions of the results of the two algorithms using IoU according to Eq. (1). If the value of the overlapping IoU is higher than a threshold that is determined in advance, then the system accepts the result region from POLO as the new template image for ATM3D. This region is extracted from the current image with the result position from POLO. In other words, the controller continues to accept the result of POLO as the next template image of the ATM3D result as long as POLO can detect the target machine accurately. Finally, the camera platform control PC determines the direction of movement of the camera and sends commands to the camera module. In our experiments, we defined the center area of the surveillance camera image with a certain margin and the system set the moving direction of the camera to bring the center point of the target object region into the center area of the image. Therefore, if the object detection area of the target construction machine is located in the upper-left corner of the image, the center point of the camera moves toward the upper-left corner to follow the center point of the object bounding box. As a result, the center point of the target object moves into the center of the camera region. The ATCS automatically controls the camera platform by iteratively performing this procedure.

3. Experiments Using ATM3D and POLO

In this section, we discuss the experimental results of the two target object detection algorithms, namely the simple image-processing-based algorithm ATM3D and the machine-learning-based algorithm POLO.

3.1. Experimental Results for ATM3D

We evaluated the detection accuracy of the proposed object detection method ATM3D. **Fig. 9** presents the experimental results for evaluating the object detection capabilities of ATM3D using a radio-control toy dump truck.

The upper photos in **Fig. 9** show the results of applying the conventional template matching method for detecting the target dump truck. In this case, the same template image is used for all matching operations and the detection of the target object fails quickly when the size of the target changes in the images.

The middle photos in **Fig. 9** present the results of applying the auto-update template matching method for detecting the target dump truck. In this case, the template image is updated for each matching operation and the detection of the target object succeeds, even when the size of the target object changes in the images. However, the template image does not cover all appearances of the dump truck, so it is easy to fail to detect the target object if the appearance of the target changes.

The lower photos in **Fig. 9** present the results of applying ATM3D, which is a combination of the auto-update template matching method and motion detection method, for detecting the target dump truck. In this case, the template image is updated properly for each matching operation and the detection of the target object succeeds to the end of the sequence. The template images cover all appearances of the dump truck, so this object detection method is robust, even when the size of the target changes.

Based on these experimental results, we have confirmed that the proposed ATM3D method, which combines auto update template matching and motion detection, is a robust object detection and tracking method.

Table 1. Results of different methods (average precision).

Models	Dump track [%]	Excavator [%]
SSD512	85.2	79.6
YOLOv2	84.5	78.2
POLO	89.1	86.4

3.2. Experimental Results for POLO

We quantified the detection accuracy of the proposed object detection method POLO. We trained models using the training dataset from the unmanned construction dataset and evaluated the proposed model using the testing dataset. As an evaluation metric, we adopted the average precision defined in PASCAL VOC [22], which is commonly used for evaluating the performance of object detection methods.

Compared to off-the-shelf one-stage detectors such as SSD and YOLOv2, which are typically able to process in real time, our proposed part-based detector significantly improves the detection results. We trained SSD and YOLOv2 models to detect a construction machine using only labels of complete appearances based on the training procedures presented in [13, 18]. All detectors, including POLO, were trained to detect two types of construction machines: dump trucks and excavators.

Comparison results for the average precision over an IoU of 0.5 are presented in **Table 1**. According to the detection results for both construction machines, our method obtains improved detection results compared to the backbone detection model of YOLOv2. The proposed method increases performance by 4.6 points on the dump trucks and 8.2 points on the excavators when compared to the baseline. More importantly, compared to off-the-shelf detectors, we obtained superior performance on the excavators in the dataset. Excavators have more widely varying appearances than dump trucks based on their movement and work operations. For example, the appearances of excavators that extend and fold their arms are not similar. Therefore, typical detectors suffer from appearance changes, whereas the part-based detector can still detect changing targets. Additionally, the bounding boxes of excavators tend to have a certain margin because their arms and booms move widely, whereas those of dump trucks are largely filled by their appearances. It is assumed that these margins also affect the object detection performances of off-the-shelf detectors.

Qualitative detection results of our proposed detector were selected randomly from the evaluation dataset, as shown in **Fig. 10**. The predicted bounding boxes are plotted with different grayscale levels. One can see that the proposed method can detect target objects robustly, even if their appearances overlap with each other. We confirmed that it is even robust in some occlusion scenes, such as **Figs. 10(a)** and **(b)**.

(a)

(b)

Fig. 10. Visualization results for POLO detection.

Fig. 11. Experimental field of an unmanned construction site.

4. Real Field Experiments

The effectiveness of the proposed ATCS under real field conditions is evaluated in this section. The setup for the corresponding experiment is summarized in Section 4.1. Comparisons between the results of each method are presented in Section 4.2. The experimental results of ATCS are discussed in Section 4.3, including a comparison to other off-the-shelf algorithms for tracking objects and an issue related to our integration method.

4.1. Experimental Setup

To evaluate the proposed system, we performed experiments at a real unmanned construction site. **Fig. 11** presents the experimental conditions, where the surveillance camera can see the overall field of the unmanned construction site. The experimental field is assumed to

be a general unmanned construction site and one camera follows a crawler dump truck that moves around in a field of approximately 70×60 m at a maximum speed of 11 km/h. In this experiment, we executed the ATCS on a laptop PC connected to a GPU server through a mobile network. ATCS performance depends on the speed of the network. Therefore, ATCS performs better on a faster network such as a local area network. However, it can also work well on mobile networks.

An evaluation dataset was prepared. It consists of a video clip captured by a real field surveillance camera and is annotated. It was used to test the comprehensive capabilities of each algorithm. This video clip shows a construction machine moving around in the experimental field for approximately 2 min and 39 s. The total number of images is 4,765 because images were acquired at 30 fps. In this video clip, the construction machine first runs straight from the back toward the camera, turns to the left of the screen, and then recedes to the right (see **Figs. 12** and **14** in the following sections). Therefore, tracking methods using off-the-shelf algorithms suffer from significant changes in the appearance of the object. The target construction machine is always captured approximately at the center of the screen because the surveillance camera tracks the construction machine. The number displayed in the upper-left corner of each image indicates the frame number in the video clip. In the second half of the video, another crawler dump truck appears in the foreground, so it is necessary to distinguish between the two vehicles and accurately capture and follow the target object.

4.2. Experimental Results for Each Method

In this section, we discuss the evaluation results of both object detection methods (i.e., ATM3D and POLO) used in our proposed system. To evaluate their tracking abilities, we applied these methods directly to the testing dataset. In other words, each method was used to detect the target object in each frame. We evaluated these tracking methods by introducing two evaluation indexes: the trackable time ratio and the detection ratio. First, we introduce the trackable time ratio, which indicates how long the target object is tracked, because it is significant to track a target object continuously in an auto-tracking system. The trackable time ratio is defined as the length of time each method can track the target object in a test video clip without detaching the annotated bounding boxes over the total duration of the test video clip. It is calculated by dividing the number of frames from the start of following the target object until it can no longer be followed by the number of frames in the entire evaluation dataset. The first frame that cannot be tracked is the frame in which the IoU of the ground truth and detection result becomes zero for the first time. Additionally, we evaluated the detection recall ratio. This evaluation index represents how well each algorithm can detect the target object with high localization accuracy. The detection ratio is calculated by dividing the number of counted frames with the detected area by the total number of frames. Here, the detected area is defined as the IoU, which is the degree of overlap between the correct and detected target object areas and must be greater than 0.7. This threshold value is sufficient to distinguish the target object from other objects. This evaluation highlights the localization ability and robustness of each method for object detection. If this value is high, it is considered that the method is able to detect an object with a bounding box that is close to the annotation data. By evaluating the trackable time ratio and detection ratio together, we quantitatively evaluated the performance of each tracking algorithm for detecting and tracking the target object correctly. The quantitative results for these values for both methods are presented in **Table 2**. These results demonstrate that both methods can track the target object across the evaluation dataset and from the results of the detection ratio, one can see that POLO has a higher position detection accuracy than ATM3D. The total time required for each method to detect the target object is approximately 0.03–0.1 s for ATM3D and approximately 0.5–2 s for POLO.

Table 2. Comparison of the two methods composing ATCS.

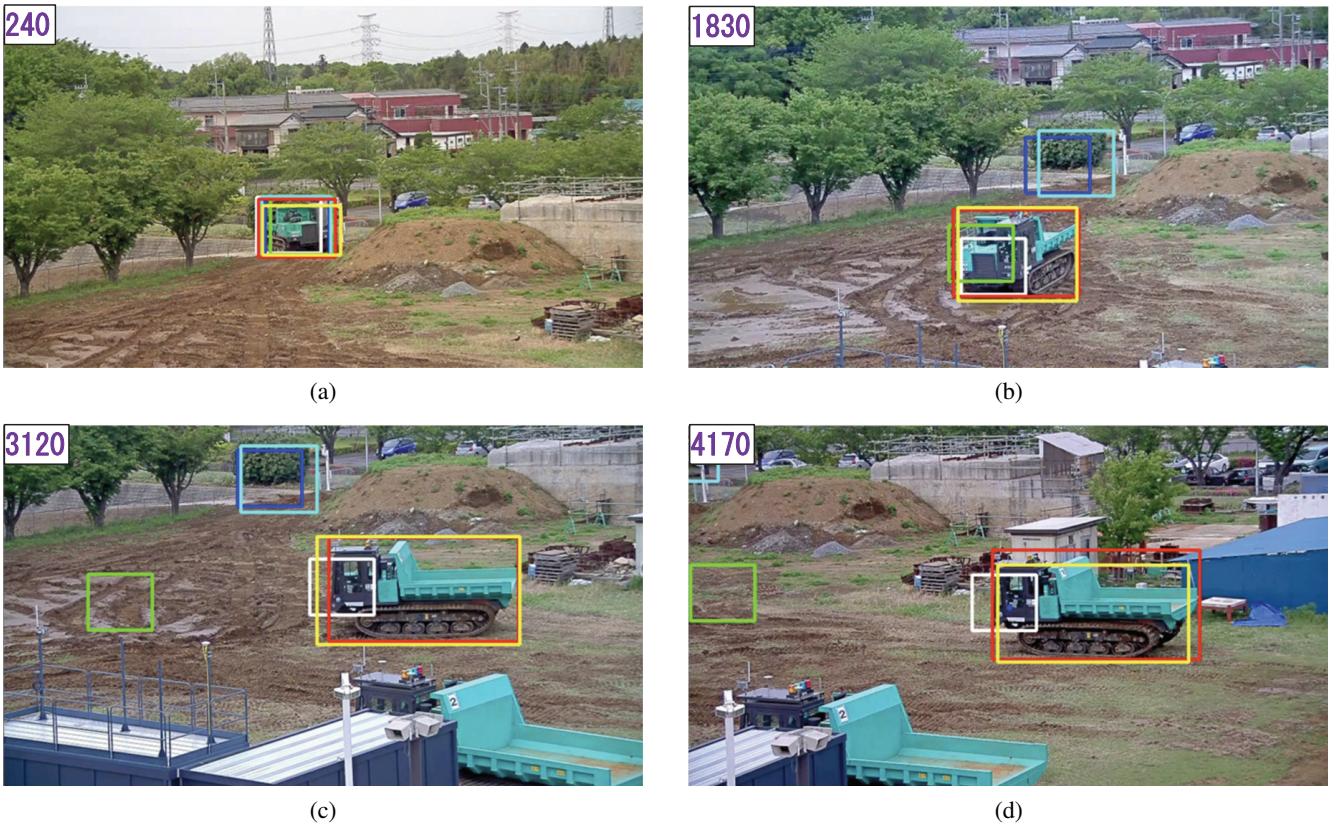
Method	Trackable time ratio [%]	Detection ratio [%]
ATM3D	100	62.58
POLO	100	84.28

Table 3. Evaluation results of ATCS and off-the-shelf tracking algorithms.

Methods	Trackable time ratio [%]	Detection ratio [%]
MedianFlow	37.22	10.38
KCF	35.65	7.86
MIL	56.78	9.43
Boosting	100	23.27
ATCS	100	77.67

4.3. Experimental Results for the Integrated System

In this section, we present comprehensive experiments evaluating the tracking ability of ATCS by comparing it to off-the-shelf methods such as MedianFlow. For comparison, we tested the following target object detection and tracking methods: MedianFlow [24], KCF [25], MIL [26], and Boosting [27], which are implemented in OpenCV. The results obtained for both the trackable time ratio and detection ratio are presented in **Table 3**. These quantitative results demonstrate that both Boosting and our ATCS are able to follow the target object continuously. In contrast, some of the popular tracking methods fail to track the target continuously. Regarding the detection ratio, Boosting yields the best results among the off-the-shelf algorithms. However, one can see that our



(a) (b) (c) (d)

Fig. 12. Qualitative results of each method (red: ground truth, white: Boosting, green: MIL, blue: KCF, light blue: MedianFlow, yellow: ATCS).

system has an advantage of 54.4 points compared to the Boosting method.

When comparing the results in **Tables 2** and **3**, the detection ability of ATCS is greater than that of ATM3D. However, the detection ratio of ATCS is lower than that of POLO. This issue stems from the integration of ATM3D and POLO, as discussed in Section 5.1. Qualitative detection results are presented in **Fig. 12**. The red bounding box represents the ground truth and the white, green, blue, light blue, and yellow bounding boxes represent the results of Boosting, MIL, KCF, MedianFlow, and ATCS, respectively. In **Fig. 12(a)**, all methods track the target object successfully because its appearance is similar to that of the initial image. However, **Figs. 12(b)–(d)** reveal that some tracking algorithms fail to follow the target because the appearance of the target machine in each frame differs from the initial image. In these results, one can see that only ATCS (yellow) and Boosting (white) are able to follow the target object until the evaluation dataset ends. From **Figs. 12(c)** and **(d)**, one can see that the Boosting (white) result fits the region of the cab and suffers from significant changes in the object's appearance. MIL (green) succeeds in tracking until the target construction machine turns and fails when the appearance of the machine changes completely.

5. Improved System

5.1. Issues and an Improved Integration Method

Based on the experimental results shown in Section 4, we confirmed that the detection ratio of ATCS is lower than that of POLO, so we analyzed this issue further. We identified the following reasons for this issue: changing the direction of the camera platform changes the results of POLO. In other words, when the camera unit moves to follow the object, even if POLO returns the correct location of the object, there is a difference between the current position of the object and the result. This caused less overlap area between the ground-truth and detected areas. Therefore, the detection rate decreased. **Fig. 13** presents a visualization of this issue. ATCS utilizes only the returned position data from POLO, so when moving the camera platform, the result of POLO, which represents the detected target object position in the image before the camera platform moves, does not match the current position of the object. A low-speed network may also cause problems similar to this issue. However, we could not confirm issues caused by the low-speed network in this experiment.

To improve the detection accuracy and develop a more robust tracking system, we propose a new integration method that utilizes the template matching method. We added an algorithm that crops the target object image from a sent image and finds the matching area in the current im-

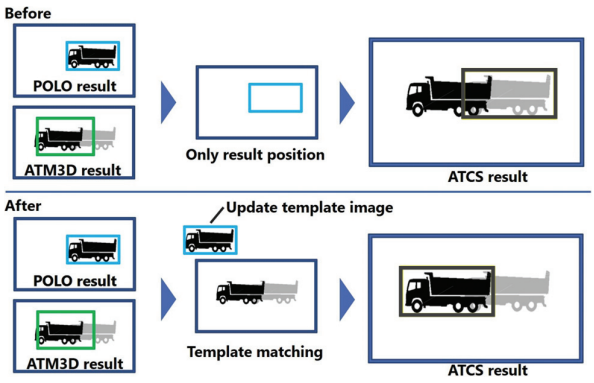
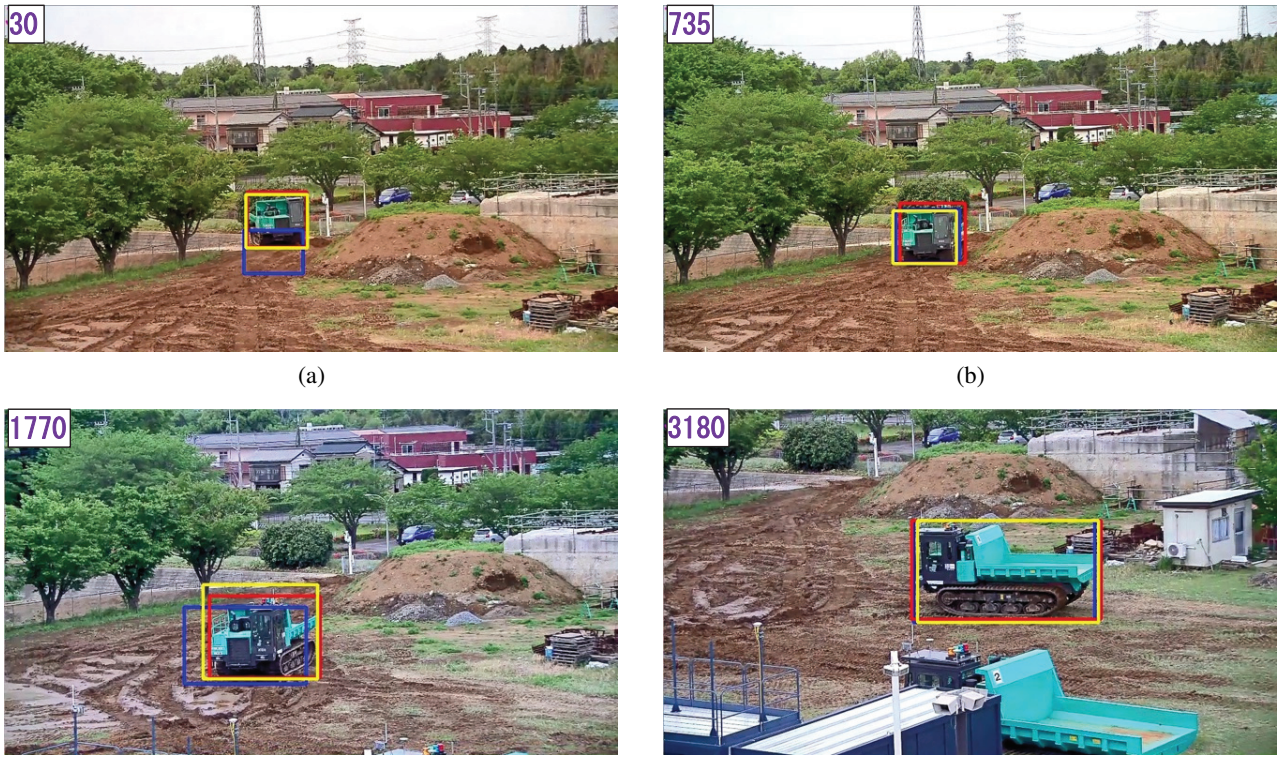


Fig. 13. Issue associated with moving the camera.

Table 4. Comparison of ATCS and ATCS+.

Method	Trackable time ratio [%]	Detection ratio [%]
ATCS	100	77.67
ATCS+	100	82.70



(a) (b) (c) (d)

Fig. 14. Comparison of ATCS and ATCS+ (red: ground truth, blue: ATCS, yellow: ATCS+).

age. In other words, after receiving the detected region of the target construction machine, it extracts the detected region from the sent image. Therefore, the template image represents the construction machine with the accuracy of POLO. Next, the algorithm finds the area in the current frame that matches the template image. This method is robust to shifts in the camera image. After finding the corresponding area in the template image, the algorithm returns its position. The post-processing of the system after obtaining the value from POLO is the same. Therefore, ATCS combines the results of the new algorithm with the results of ATM3D. We call this new integration method ATCS+. **Fig. 13** presents a visualization of the new integration method.

5.2. Comparison of ATCS and ATCS+

In this section, we compare the experimental results for ATCS and ATCS+. **Table 4** presents the detection ratios and trackable time ratios of both methods under the same conditions as the experiment described in Section 4.3. Both are able to track the target object continuously. Regarding the detection ratio, ATCS+ exhibits performance improvements of approximately five points compared to ATCS. In the evaluation dataset, there are a few shifts in the camera image. If there are more changes in the camera direction in other datasets or real field videos, the difference between the detection ratios of the two methods should expand. **Fig. 14** presents a qualitative comparison of the two methods. The red bounding boxes represent the ground truth and the blue and yellow bounding

boxes represent the results of ATCS and ATCS+, respectively. In particular, **Figs. 14(a)** and **(c)** represent the time at which the shift in the camera's direction occurs. During our experiments, it was clear that the new integration approach has very desirable detection profiles, particularly when the camera image changes.

In this study, YOLOv2 was used as a baseline for the construction of POLO, but because the performances of its successors YOLOv3 to YOLOv5 are improved, it is thought that the recognition accuracy of each component will increase and the recognition ability for the entire construction machine will also be improved by configuring POLO based on these new versions. Because POLO is constructed as an occlusion-aware method, it is expected that the proposed method and system will become more occlusion-resistant by improving the performance of the baseline.

6. Conclusion

In this study, we solved the problems that arise in constructing a robust object tracking system for an unmanned construction site and successfully developed a novel system integrating two different types of algorithms based on ordinary image processing and machine learning. Regarding the detection algorithm using simple image processing, our proposed ATM3D method is able to continue tracking for a longer time than other methods. Regarding the detection algorithm based on machine learning, our proposed POLO model, which expands from an existing neural network model, detects construction machines robustly.

Real-world experiments demonstrated that the proposed methods are accurate and robust while maintaining practical real-time processing efficiency. We also studied issues caused by changing the camera direction and/or network delay. Our improved object tracking method, called ATCS+, provides robust tracking features than the original method.

Acknowledgements

This study was conducted with the aid of an FY2016 Research and Development Grant from the Japan Construction Machinery and Construction Association, and field experiments were conducted with the cooperation of Kumagai Gumi Co., Ltd.

References:

[1] Unmanned Construction Association, "Trends and prospects for unmanned construction," Construction Project Planning, Vol.681, pp. 6-12, 2006 (in Japanese).

[2] K. Chayama et al., "Technology of Unmanned Construction System in Japan," J. Robot. Mechatron., Vol.26, No.4, pp. 403-417, doi: 10.20965/jrm.2014.p.0403, 2014.

[3] T. Bock, "Construction Robotics," J. Robot. Mechatron., Vol.28, No.2, pp. 116-122, doi: 10.20965/jrm.2016.p0116, 2016.

[4] K. Tateyama, "Achievement and Future Prospects of ICT Construction in Japan," J. Robot. Mechatron., Vol.28, No.2, pp. 123-128, doi: 10.20965/jrm.2016.p0123, 2016.

[5] T. Tanimoto et al., "Research on Superimposed Terrain Model for Teleoperation Work Efficiency," J. Robot. Mechatron., Vol.28, No.2, pp. 173-184, doi: 10.20965/jrm.2016.p0173, 2016.

[6] T. Nagano et al., "Arbitrary Viewpoint Visualization for Teleoperated Hydraulic Excavators," J. Robot. Mechatron., Vol.32, No.6, pp. 1223-1243, doi: 10.20965/jrm.2020.p1233, 2020.

[7] M. Ito, Y. Funahara, S. Saiki, Y. Yamazaki, and Y. Kurita, "Development of a Cross-Platform Cockpit for Simulated and Tele-Operated Excavators," J. Robot. Mechatron., Vol.31, No.2, pp. 231-239, doi: 10.20965/jrm.2019.p0231, 2019.

[8] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," ISPRS J. of Photogrammetry and Remote Sensing, Vol.117, pp. 11-28, doi: 10.1016/j.isprsjprs.2016.03.014, 2016.

[9] R. Brunelli, "Template Matching Techniques in Computer Vision: Theory and Practice," John Wiley and Sons, 2009.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Communications of the ACM, Vol.60, Issue 6, 2012.

[11] Y. LeCun, B. Boser, J. S. Denker et al., "Backpropagation applied to handwritten zip code recognition," J. of Neural Computation, Vol.1, Issue 4, pp. 541-551, doi: 10.1162/neco.1989.1.4.541, 1989.

[12] J. Redmon, S. Divvala, R. Girshick et al., "You only look once: Unified, real-time object detection," Proc. of 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 779-788, 2016.

[13] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," Proc. of 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 6517-6525, 2017.

[14] M. Inoue and T. Yoshimi, "Automatic Tracking Camera System for Construction Machines by Combined Image Processing," Proc. of the 35th Int. Symp. on Automation and Robotics in Construction (ISARC 2018), pp. 630-636, doi: 10.22260/ISARC2018/0086, 2018.

[15] A. A. Malik, A. Khalil, and H. U. Khan, "Object Detection and Tracking using Background Subtraction and Connected Component Labelling," Int. J. of Computer Applications, Vol.75, No.13, doi: 10.5120/13168-0421, 2013.

[16] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," Proc. of 2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 2155-2162, 2014.

[17] M. Fujitake and T. Yoshimi, "Estimation System of Construction Equipment from Field Image by Combination Learning of Its Parts," Proc. of the 11th Asian Control Conf. (ASCC), pp. 1672-1676, doi: 10.1109/ASCC.2017.8287425, 2017.

[18] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multibox detector," Proc. of the 14th European Conf. on Computer Vision (ECCV), B. Leibe, J. Matas, N. Sebe, and M. Welling (Eds.), "Computer Vision – ECCV 2016," Springer, pp. 21-37, doi: 10.1007/978-3-319-46448-0_2, 2016.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proc. of 2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 580-587, 2014.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real time object detection with region proposal networks," Proc. of IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.39, No.6, pp. 1137-1149, doi: 10.1109/TPAMI.2016.2577031, 2015.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. of 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[22] M. Everingham, S. M. A. Eslami, L. V. Gool et al., "The Pascal Visual Object Classes Challenge: A Retrospective," Int. J. of Computer Vision, Vol.111, Issue 1, pp. 98-136, doi: 10.1007/s11263-014-0733-5, 2015.

[23] O. Russakovsky, J. Deng, H. Su et al., "ImageNet Large Scale Visual Recognition Challenge," Int. J. of Computer Vision, Vol.115, Issue 3, pp. 211-252, doi: 10.1007/s11263-015-0816-y, 2015.

[24] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-Backward Error: Automatic Detection of Tracking Failures," Proc. of the 20th Int. Conf. on Pattern Recognition, doi: 10.1109/ICPR.2010.675, pp. 2756-2759, 2010.

[25] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters," Proc. of IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.37, No.3, pp. 583-596, doi: 10.1109/TPAMI.2014.2345390, 2015.

[26] B. Babenko, M.-H. Yang, and S. Belongie, "Visual Tracking with Online Multiple Instance Learning," Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), doi: 10.1109/CVPR.2009.5206737, pp. 983-990, 2009.

[27] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," Proc. of the 17th British Machine Vision Association, pp. 6.1-6.10, doi: 10.5244/C.20.6, 2006.

Name:
Masato Fujitake

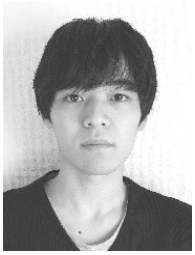
Affiliation:
Doctoral Course Student, Department of Informatics, The Graduate University for Advanced Studies (SOKENDAI) (Currently)

Address:
Shonan Village, Hayama, Kanagawa 240-0193, Japan

Brief Biographical History:
2017- Master Course Student, Graduate School of Engineering and Science, Shibaura Institute of Technology
2019- Doctoral Course Student, Department of Informatics, SOKENDAI

Main Works:
• "Temporally-aware Convolutional Block Attention Module for Video Text Detection," IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), 2021.
• "Real-time Object Detection by Feature map forecast For Live Streaming Video," Proc. of IEEE Int. Conf. on Multimedia and Expo (ICME), pp. 1-6, 2021.
• "Temporal Feature Enhancement Network with External Memory for Object Detection in Surveillance Video," Proc. of the 25th Int. Conf. on Pattern Recognition (ICPR), pp. 7684-7691, 2020.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)
• The Computer Vision Foundation (CVF)



Name:
Makito Inoue

Affiliation:
Research and Development Division, Denso Wave Inc. (Currently)

Address:
1 Yoshiike, Kusagi, Agui-cho, Chita-gun, Aichi 470-2297, Japan

Brief Biographical History:
2017- Master Course Student, Graduate School of Engineering and Science, Shibaura Institute of Technology
2019- Denso Wave Inc.

Main Works:
• "Automatic Tracking Camera System for Construction Machines by Combined Image Processing," Proc. of the 35th Int. Symp. on Automation and Robotics in Construction (ISARC 2018), pp. 630-636, 2018.

Name:
Takashi Yoshimi

Affiliation:
Professor, Department of Electrical Engineering, College of Engineering, Shibaura Institute of Technology

Address:
3-7-5 Toyosu, Koto-ku, Tokyo 135-8548, Japan

Brief Biographical History:
1987- Toshiba Corporate Research and Development Center
1999-2001 Visiting Researcher, Japan Atomic Energy Research Institute
2000 Received Ph.D. in Mechanical Engineering from Osaka University
2009- Professor, Department of Electrical Engineering, Shibaura Institute of Technology

Main Works:
• Robot systems, force control, task planning, practical robot technologies

Membership in Academic Societies:
• The Japan Society of Mechanical Engineers (JSME)
• The Society of Instrument and Control Engineers (SICE)
• The Robotics Society of Japan (RSJ)
• The Institute of Electrical and Electronic Engineers (IEEE)